

In the claims:

Please cancel claims 1-20 without prejudice.

Please add the following new claims:

- 54531
AC
- 1 21. (New) A method comprising:
2 generating a first test program to test the functionality of an integrated circuit
3 (IC);
4 executing the first test program;
5 determining whether the IC has been sufficiently tested; and
6 if not, determining whether a predetermined test program population threshold
7 has been reached.
- 1 22. (New) The method of claim 21, further comprising:
2 generating a second test program if the predetermined test program population
3 threshold has been reached; and
4 executing the second test program.
- 1 23. (New) The method of claim 22, wherein generating the first test program
2 comprises:
3 generating a first abstract syntax tree (AST);
4 generating a first set of instructions and data for the first AST; and
5 translating the first AST into a first executable test program.
- 1 24. (New) The method of claim 23, wherein generating the second test program
2 comprises:
3 generating a second abstract syntax tree (AST);
4 generating a second set of instructions and data for the second AST; and
5 translating the second AST into a second executable test program.

1 25. (New) The method of claim 24, further comprising mutating a selected AST.

1 26. (New) The method of claim 25, wherein mutating a selected AST comprises:

2 selecting an AST;

3 removing a segment of the selected AST; and

4 inserting a replacement segment into the selected AST to form a mutated AST.

1 27. (New) The method of claim 26, further comprising:

2 generating a third set of instructions and data for the mutated AST; and

3 translating the mutated AST into a third executable test program.

1 28. (New) The method of claim 25, wherein mutating a selected AST comprises:

2 selecting the first AST and the second AST; and

3 combining a segment of the first AST with a segment of the second AST to form

4 a mutated AST.

1 29. (New) The method of claim 28, further comprising:

2 generating a third set of instructions and data for the mutated AST; and

3 translating the mutated AST into a third executable test program.

1 30. (New) The method of claim 21, further comprising:

2 adding the first AST and corresponding coverage data into test program

3 population after the first test program has been executed.

1 31. (New) A computer system comprising:

2 a storage device coupled to a processor and having stored therein at least one

3 routine, which when executed by the processor, causes the processor to generate data, the

4 routine causing the processor to,

5 generate a first test program to test the functionality of an integrated circuit (IC);

6 execute the first test program;
7 determine whether the IC has been sufficiently tested; and
8 determine whether a predetermined test program population threshold has been
9 reached.

1 32. (New) The computer system of claim 31, wherein the routine further causes the
2 processor to,
3 generate a second test program if the predetermined test program population
4 threshold has been reached; and
5 execute the second test program.

1 33. (New) The computer system of claim 32, wherein generating the first test
2 program comprises:
3 generating a first abstract syntax tree (AST);
4 generating a first set of instructions and data for the first AST; and
5 translating the first AST into a first executable test program.

1 34. (New) The computer system of claim 33, wherein generating the second test
2 program comprises:
3 generating a second abstract syntax tree (AST);
4 generating a second set of instructions and data for the second AST; and
5 translating the second AST into a second executable test program.

1 35. (New) The computer system of claim 34, wherein the routine further causes the
2 processor to mutate a selected AST.

1 36. (New) The computer system of claim 35, wherein mutating a selected AST
2 comprises:
3 selecting an AST;

4 removing a segment of the selected AST; and
5 inserting a replacement segment into the selected AST to form a mutated AST.

1 37. (New) The computer system of claim 36, wherein the routine further causes the
2 processor to,
3 generate a third set of instructions and data for the mutated AST; and
4 translate the mutated AST into a third executable test program.

1 38. (New) The computer system of claim 35, wherein mutating a selected AST
2 comprises:
3 selecting the first AST and the second AST; and
4 combining a segment of the first AST with a segment of the second AST to form
5 a mutated AST.

1 39. (New) The computer system of claim 38, wherein the routine further causes the
2 processor to,
3 generating a third set of instructions and data for the mutated AST; and
4 translating the mutated AST into a third executable test program.

54B3
1 40. (New) The computer system of claim 31, wherein the routine further causes the
2 processor to,
3 adding the first AST and corresponding coverage data into test program
4 population after the first test program has been executed.

1 41. (New) A validation test system comprising:
2 a test builder to generate test programs to test the functionality of an integrated
3 circuit (IC);
4 a test generator to translate the test programs into an executable test;
5 a test analyzer to execute the test programs; and

6 a feedback engine to build a test population of test programs by generating an
7 abstract syntax tree (AST) for each test program.

1 42. (New) The system of claim 41, wherein the feedback engine determines whether
2 a predetermined test program population threshold has been reached after a test program
3 has been executed.

1 43. (New) The system of claim 42, wherein the feedback engine generates one or
2 more mutated ASTs if it is determined that the predetermined test program population
3 threshold has been reached.

1 44. (New) The system of claim 43, wherein the feedback engine generates a mutated
2 AST by selecting a first AST, removing a segment of the first AST and inserting a
3 replacement segment into the first AST.

1 45. (New) The system of claim 43, wherein the feedback engine generates a mutated
2 AST by selecting a first AST and a second AST and combining a segment of the first
3 AST with a segment of the second AST to form.
